

SISTEM INFORMASI PENGISIAN NILAI BERBASIS JAVA WEB MENGGUNAKAN LOCAL SERVER PADA SMK 2 MEI BANDAR LAMPUNG

Ferly Ardy

*Program Studi Sistem Informasi STMIK DCC Lampung
Jl. Cut Nyak Dien No. 65 Durian Payung (Palapa) Bandar Lampung
E-mail: ferly@dcc.ac.id*

ABSTRAK

SMK 2 Mei Bandar Lampung yang mengusung konsep sekolah terpadu yang telah membuka 11 program keahlian. Pengelolaan data nilai adalah hal terpenting di setiap sekolah. Setiap habis semester para guru harus memberikan atau mengumpulkan nilai kepada staf TU untuk dikelola. Dimana nilai tersebut akan di distribusikan kepada siswa pada setiap semester. Namun untuk pengisian nilai saat ini pengolahan datanya masih dilakukan secara konvensional, yakni masih menggunakan sistem *Excell* yang kemudian ini membuat kesulitan petugas untuk mengelola data yang sudah diinput karena keterbatasan sistem untuk mensortir informasi data nilai sesuai kebutuhan. Selain itu penginputan data nilaipun sering terjadi kesalahan dalam terjadi duplikat nama nilai dan lain-lain.

Dengan permasalahan tersebut peneliti melakukan sebuah penelitian tentang pengisian nilai raport pada SMK 2 Mei Bandar Lampung. Dengan menggunakan metode extreme programming dan alat bantu pengembangan sistem berbasis *UML (Unified Modelling Language)*, pengumpulan data dengan teknik berupa observasi, wawancara, studi pustaka, analisis sistem (*activity diagram*), desain sistem (*use case diagram*) perancangan database (*class diagram*), dan pembuatan sistem informasi pada pengisian nilai raport pada SMK 2 Mei Bandar Lampung Berbasis *Java Web*.

Hasil akhir dari penelitian ini adalah sebuah Sistem informasi pengisian nilai raport pada SMK 2 Mei Bandar Lampung Berbasis *Java Web* sehingga dapat lebih baik dalam mengelola data nilai, sehingga guru, siswa dan bahkan orang tua bisa berinteraksi langsung dengan sistem dengan tujuan informasi data nilai dapat terdistribusi dengan baik.

Kata kunci : *extreme programming, UML (Unified Modelling Language), Activity Diagram, Use Case Diagram, Java Web, Pengisian Nilai*

1. PENDAHULUAN

Sekolah Menengah Kujuran (SMK) adalah salah satu bentuk satuan pendidikan formal yang menyelenggarakan pendidikan kejuruan pada jenjang pendidikan menengah sebagai lanjutan dari SMP/MTs atau bentuk lain yang sederajat atau lanjutan dari hasil belajar yang diakui sama/setara SMP/MTs. Untuk itu pendidikan formal yang baik adalah harus memiliki tata kelola IT yang baik khususnya untuk pengolahan data nilai siswa.

Dalam upaya untuk meningkatkan layanan terhadap siswanya, SMK 2 Mei Bandar Lampung membutuhkan sebuah sistem pengelolaan data dilai yang terintegrasi antara guru, wali kelas, orang tua dan tentunya siswa. Dengan adanya kondisi tersebut disini peneliti akan membangun sebuah sistem pengelolaan data nilai dengan judul Sistem Informasi Pengisian Nilai Berbasis Java Web Menggunakan Local Server Pada Smk 2 Mei Bandar Lampung yang nantinya sistem tersebut akan sangat membantu para guru untuk melakukan input data secara langsung.

1.1 KAJIAN PUSTAKA

1.1.1 Metode Pengembangan Sistem

Extreme Programming (XP) merupakan salah satu metodologi dalam rekayasa perangkat lunak dan juga merupakan satu dari beberapa *agile software development methodologies* yang berfokus pada *coding* sebagai aktivitas utama di semua tahap pada siklus pengembangan perangkat lunak (*software development lifecycle*). Metodologi ini mengedepankan proses pengembangan yang lebih responsive terhadap kebutuhan customer ("agile") dibandingkan dengan metode-metode tradisional sambil membangun suatu software dengan kualitas yang lebih baik.

Ini adalah konsep dasar dari XP yang sangat minimum dokumentasi.

1. *Extreme Programming* yang selanjutnya disingkat dengan XP merupakan salah satu dari sekian banyaknya metodologi dalam rekayasa perangkat lunak dan juga merupakan bagian dari metodologi pengembangan perangkat lunak agile.

2. Secara umum *Extreme Programming* (XP) dapat dijabarkan sebagai sebuah pendekatan pengembangan perangkat lunak

yang mencoba meningkatkan efisiensi dan fleksibilitas dari sebuah proyek pengembangan perangkat lunak dengan mengkombinasikan berbagai ide simpel/ sederhana tanpa mengurangi kualitas software yang akan dibangun.

3. XP dikembangkan oleh Beck, Cunningham, dan Jeffries dan ini merupakan light weight discipline pengembangan perangkat lunak berdasarkan empat core value.

1.2 ALAT PENGEMBANGAN SISTEM

1.2.1 UML (*Unified Modeling Language*)

"UML (*Unified Modeling Language*) adalah Metodologi kolaborasi antara metoda-metoda Booch, OMT (*Object Modeling Technique*), serta OOSE (*object Oriented Software Engineering*) dan beberapa metoda lainnya, merupakan metodologi yang paling sering digunakan saat ini untuk analisa dan perancangan sistem dengan metodologi berorientasi objek mengadaptasi maraknya penggunaan bahasa "pemrograman berorientasi objek" (OOP)".

Berdasarkan pendapat yang dikemukakan di atas dapat ditarik kesimpulan bahwa "*Unified Modelling Language (UML)* adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasikan, menspesifikasikan, membangun dan pendokumentasian dari sebuah sistem pengembangan perangkat lunak berbasis Objek (*Object Oriented programming*)". (Nugroho 2012:4)

Langkah-langkah penggunaan *Unified Modeling Language (UML)*

Menurut Henderi (2013:6), langkah-langkah penggunaan *Unified Modeling Language (UML)* sebagai berikut:

1. Buatlah daftar *business process* dari *level* tertinggi untuk mendefinisikan aktivitas dan proses yang mungkin muncul.
2. Petakan *use case* untuk setiap *business process* untuk mendefinisikan dengan tepat fungsional yang harus disediakan oleh sistem, kemudian perhalus *use casediagram* dan lengkapi dengan *requirement*, *constraints* dan catatan-catatan lain.

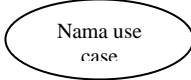
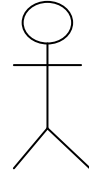

3. Buatlah *deployment diagram* secara kasar untuk mendefinisikan arsitektur fisik sistem.
4. Definisikan *requirement* lain *non fungsional*, *security* dan sebagainya yang juga harus disediakan oleh sistem.
5. Berdasarkan *use case diagram*, mulailah membuat *activity diagram*.
6. Definiskan obyek - obyek level atas *package* atau *domain* dan buatlah *sequencedan/atau collaboration* untuk tiap alur pekerjaan, jika sebuah *use case* memiliki kemungkinan alur normal dan *error*, buat lagi satu diagram untuk masing-masing alur.
7. Buatlah rancangan *user interface model* yang menyediakan antar muka bagi pengguna untuk menjalankan *skenario use case*.
8. Berdasarkan model-model yang sudah ada, buatlah *class diagram*. Setiap *package* atau *domian* dipecah menjadi *hirarki class* lengkap dengan *atribut* dan *metodenya*. Akan lebih baik jika untuk setiap *class* dibuat *unit test* untuk mengujifungsionalitas *class* dan interaksi dengan *class* lain.
9. Setelah *class diagram* dibuat, kita dapat melihat kemungkinan pengelompokkan *class* menjadi komponen-komponen karena itu buatlah *component diagram* pada tahap ini. Juga, definisikan *test integrasi* untuk setiap komponen meyakinkan ia bereaksi dengan baik.
10. Perhalus *deployment diagram* yang sudah dibuat. Detilkan kemampuan dan *requirement* piranti lunak, sistem operasi, jaringan dan sebagainya. Petakan komponen ke dalam *node*.
11. Mulailah membangun sistem. Ada dua pendekatan yang tepat digunakan:
 - 1) Pendekatan *use case* dengan mengassign setiap *use case* kepada tim pengembang tertentu untuk mengembangkan unit kode yang lengkap dengan *test*.
 - 2) Pendekatan komponen yaitu mengassign setiap komponen kepada tim pengembang tertentu.

1.2.2 Use Case Diagram

"Use Case diagram adalah representasi dari interaksi pengguna dengan sistem dan menggambarkan spesifikasi dari kasus penggunaan. Sebuah *diagram use case* dapat

menggambarkan berbagai jenis pengguna sistem dan berbagai cara yang mereka berinteraksi dengan sistem. Jenis diagram biasanya digunakan dalam hubungannya dengan kasus penggunaan tekstual dan akan sering disertai dengan jenis lain dari diagram juga."









Tabel 1 Use Case Diagram

Simbol	Keterangan
	<i>Use case</i> digunakan sebagai lingkaran elips dengan nama use case dituliskan didalam elips tersebut.
	Aktor adalah pengguna sistem. Aktor tidak terbatas hanya pada manusia saja, jika sebuah sistem berkomunikasi dengan aplikasi lain dan membutuhkan input atau memberikan output, maka aplikasi tersebut juga bisa dianggap sebagai aktor
	Asosiasi digunakan untuk menghubungkan aktor dengan <i>use case</i> . Asosiasi digambarkan dengan sebuah garis yang menghubungkan antara aktor dengan <i>use case</i>

1.2.3 Activity Diagram

"Diagram aktivitas adalah representasi grafis dari seluruh tahapan alur kerja. Diagram ini mengandung aktivitas, pilihan tindakan, perulangan dan hasil dari aktivitas tersebut. Pada pemodelan UML, diagram ini dapat digunakan untuk menjelaskan proses bisnis dan alur kerja operasional secara langkah demi langkah dari komponen suatu sistem.



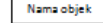

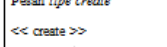
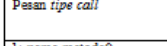
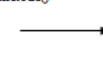
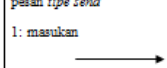
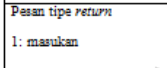
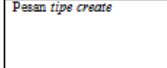
Tabel 2 Activity Diagram

Symbol	Keterangan
	Status awal aktivitas sistem sebuah diagram aktivitas memiliki sebuah status awal
	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
	Asosiasi perambatan dimana jika ada pilihan aktivitas lebih dari satu
	Asosiasi perambatan dimana lebih dari satu aktivitas digabungkan menjadi satu
	Sebuah file computer dan mana data dibaca atau direkam selama kegiatan basis
	Memiliki dokumen sumber atau laporan
	Status akhir yang dilakukan sistem sebuah diagram aktivitas memiliki sebuah status akhir
	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

1.2.4 Sequence Diagram

"Sequence diagram adalah diagram interaksi yang menunjukkan bagaimana proses beroperasi dengan satu sama lain dan dalam rangka apa. Ini adalah *konstruksi dari Bagan Message Sequence*. Sebuah diagram urutan menunjukkan interaksi objek diatur dalam urutan waktu. Ini menggambarkan objek dan kelas yang terlibat dalam skenario dan urutan pesan yang dipertukarkan antara objek yang dibutuhkan untuk melaksanakan *fungsi skenario*. *Sequence diagram* biasanya terkait dengan *realisasi use case* dalam *Logical View sistem* dalam pengembangan. *Sequence diagram* kadang-kadang disebut diagram peristiwa atau skenario acara.

Tabel 3 Sequence Diagram

Simbol	Keterangan
	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor
	Menyatakan himpunan suatu objek
	Menyatakan objek yang berisi pesan
	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan
	Menyatakan suatu objek membuat objek lain, arah panah objek yang dibuat
	Menyatakan suatu objek memanggil operasi metode
	Ada pada objek lain atau dirinya sendiri.
	Menyatakan bahwa suatu objek mengirimkan data/masukan informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian
	Menyatakan suatu objek mengakhiri objek lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy

2. METODE PENELITIAN

Metode pengumpulan data yang digunakan dalam penelitian ini berjalan adalah:

a. Studi Observasi

Tahap ini melakukan pengamatan langsung terhadap kegiatan prosedural pengolahan data nilai.

b. Studi Interview

Tahap ini mengumpulkan data secara tatap muka langsung dengan pelaku kegiatan.

c. Studi Pustaka

Tahap ini penulis menggunakan buku-buku literatur, jurnal penelitian sebagai acuan dalam penelitian.

Implmentasi Extreme Programming (XP) : Penerapan pengembangan perangkat lunak dalam Extreme Programming (XP) adalah sebagai berikut :

1. Planning

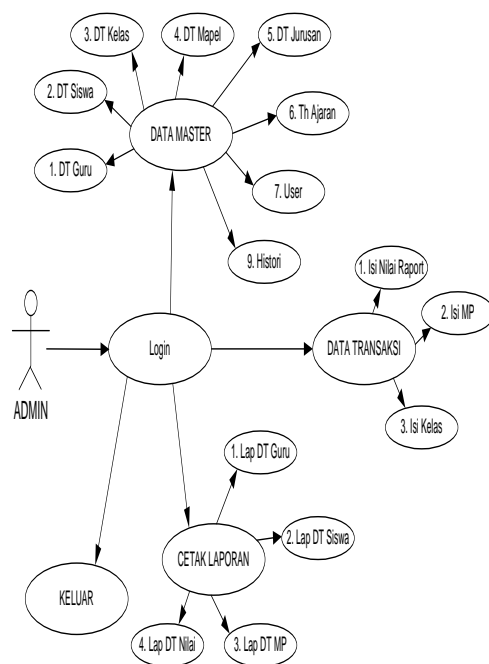
Pada tahap ini peneliti merencanakan melakukan penelitian pada lingkungan SMK 2 Mei Bandar Lampung dengan cara

berinteraksi dengan pihak terkait yang berkaitan dengan pengelolaan data nilai siswa dan kemudian mengumpulkan data-data yang sesuai dengan kebutuhan system yang akan dibuat.

2. Design

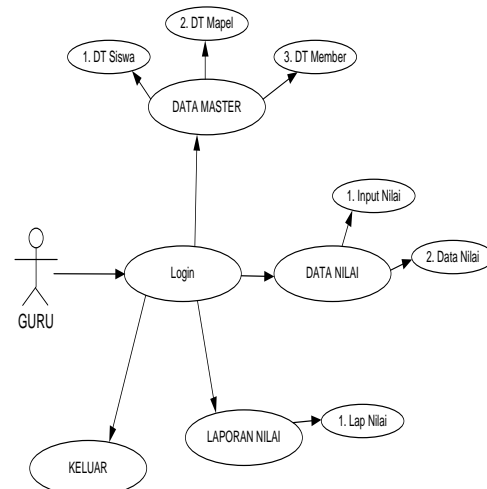
Berdasarkan rencana dan data yang telah dikumpulkan sebelumnya, pada tahap ini peneliti melakukan pembuatan rencana design sistem yang akan dibangun dengan menggunakan *tool Use Case diagram* yang terlampir pada gambar berikut :

a. Use Case Admin Pada Sistem



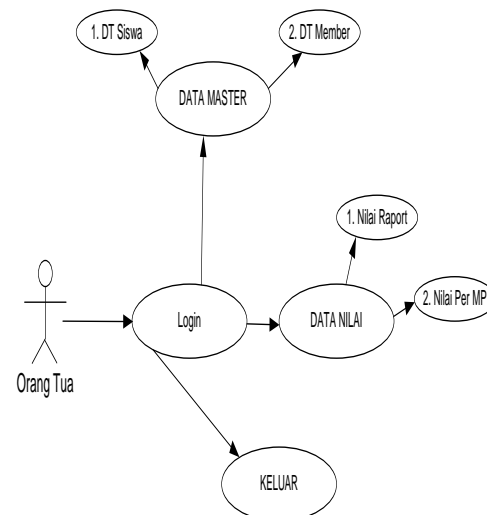
Gambar 1 Use Case Admin

b. Use Case Guru Pada Sistem



Gambar 2 Use Case Guru

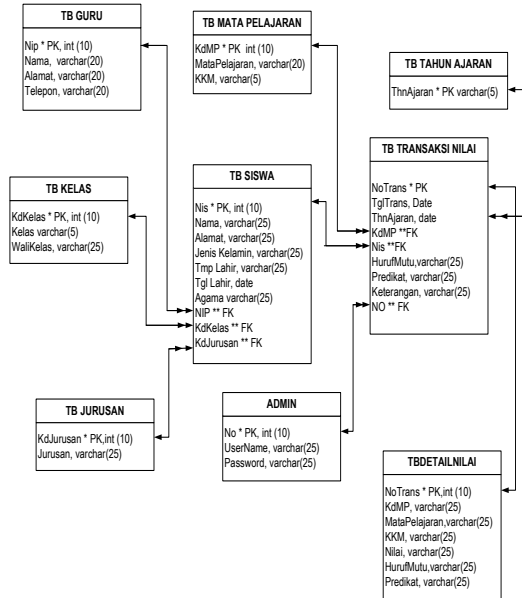
c. Use Case Orang Tua Siswa Pada Sistem



Gambar 3 Use Case Orang Tua Siswa

3. Coding

Pada tahap ini peneliti melakukan tahapan pembuatan sistem aplikasi menggunakan bahasa pemrograman Java Berbasis Web dengan berdasarkan *Use Case Sistem* (gambar 1, gambar 2 dan gambar 3) dan berdasarkan *Class Diagram* berikut :



Gambar 4 Class Diagram

4. Testing

Tahap ini adalah tahap uji coba Aplikasi Sistem Informasi Pengolahan Data Nilai, yang harapannya adalah sistem ini dapat membantu pihak sekolah dalam melakukan pengelolaan data nilai sehingga tidak terjadi lagi kesalahan-kesalahan dalam pengelolaan data nilai. Berikut adalah hasil dari pembangunan Sistem Informasi Pengolahan Data Nilai.

3. PEMBAHASAN

Halaman Menu Utama

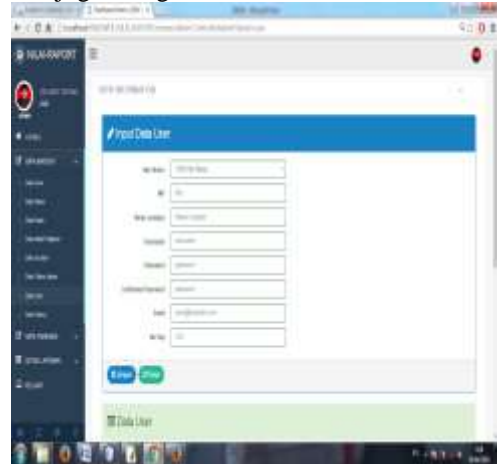
Pada halaman ini *user* harus melakukan *login* terlebih dahulu sebelum melakukan interaksi dengan sistem.



Gambar 5 Halaman Menu Utama

Halaman Menu User

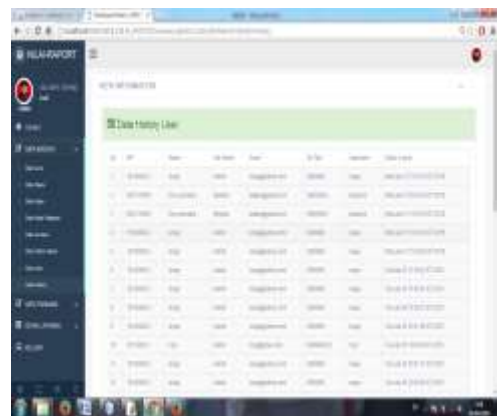
Pada halaman ini admin dapat mengelola seluruh *user* yang tersedia atau dibutuhkan, seperti guru mata pelajaran, wali kelas, siswa dan juga orang tua.



Gambar 6 Menu User

Halaman Menu History

Fungsi dari halaman menu history adalah sebuah menu yang digunakan untuk mengelola file master yang ada pada *database system* seperti pengelolaan data identitas siswa, data guru, data mata pelajaran, data jadwal, data orang tua, serta data kelas. Dan pada menu ini juga terdapat fasilitas untuk mengelola data laporan seperti, raport siswa, data guru, data mata pelajaran, serta data jadwal.



Gambar 7 Halaman Menu History

4. SIMPULAN

Kesimpulan yang didapat pada penelitian ini adalah sebagai berikut :

- a. Pembangunan Sistem Informasi Pengolahan Data Nilai Berbasis Java Web dinilai penting dikarenakan dengan adanya sistem ini akan menanggulangi permasalahan-permasalahan yang terjadi terkait data nilai pada SMK 2 Mei Bandar Lampung.
- b. Aplikasi ini diharapkan dapat menggantikan aplikasi sebelumnya yang masih menggunakan metode pengolahan data nilai sebelumnya (berbasis *excell*).

5. SARAN

Saran bagi pengembangan sistem selanjutnya adalah bahwa sistem ini masih bersifat *localhost* dan sistem ini bisa dikembangkan lagi pada tahap implementasi teknologi yang sesuai berdasarkan prioritas, sumber daya manusia serta dana yang ada.

PUSTAKA

- Djojodihardjo, Harijono. (2010: 78). Analisis dan Desain Sistem Informasi. Yogyakarta: Penerbit ANDI
- Jogiyanto. (2010:8). buku Kamus Istilah Komputer, Teknologi informasi dan Komunikasi
- Nugroho Adi. (2010:6). Metode Pengembangan Sistem UML. Yogyakarta: Penerbit Andi
- Henderi. (2013:6), Langkah-Langkah Penggunaan UML Bandung: Penerbit Andi Offset
- Subekti. (2013:1), Pengertian Perancangan Basis Data. Jakarta: Penerbit Andi Offset
- Nugroho, Adi. 2006. Rational Rose untuk Pemodelan Berorientas Objek. Bandung: Penerbit Informatika
- Nugroho, Adi. 2004. Pemrograman Berorientas Objek. Bandung: Penerbit Informatika